



Ultimate Feed Generator™

user manual

Miva Module for Miva Merchant 5

Contents

- Introduction
- Requirements
- Installation
- Usage
- Store Morph Technology
- Support

Introduction

Ultimate Feed Generator is a tool for generating product feeds and sitemaps. It can be used to create feeds for:

- Comparison Shopping Engines such as Shopzilla, NexTag, Shopping.com, PriceGrabber, etc
- Search engine sitemaps: XML Sitemaps for Google, Bing, Yahoo, and Ask.com
- HTML Store Maps: static site map page(s) on your own site that link to every (or selected) product and category

The module comes with several pre-defined feeds: an XML Sitemap, a GoogleBase template, and a feed for our FeedExact service. Feeds are defined using Store Morph Technology, the same scripting language you use to build your Miva Merchant store's pages. Ultimate Feed Generator fully supports SMT's mvt:if/mvt:else statements, mvt:foreach loops, and entities (for bringing in dynamic data).

Feeds can be generated individually, or you can select multiple feeds and run them at once. (Check them all off and go to lunch while it processes, for instance.) The output feeds can be delimited text files, XML files, HTML files, or other plain-text files.

The module has built-in support for the following:

- Custom Product Fields
- Custom Category Fields
- Attributes and Options (including Attribute Templates)
- Primary Category from our Category Breadcrumbs Plus module
- Sales Statistics
- Ratings and Review Count from Sebenza's Ultimate Reviews module
- Any style of short links, including optional the URLSTUB custom-product field format from our Smart Links for SEO module

There are also options that allow you to include or exclude products/categories based on the following rules:

- Whether the product or category is active or inactive
- Whether the product is in stock
- Minimum and maximum weight
- Minimum and maximum price
- Minimum and maximum cost
- Include or Exclude products by code (pipe-delimited list of product codes)
- Include or Exclude products by category code (pipe-delimited list of category codes)
- Include or Exclude categories by code (pipe-delimited list of category codes)

Requirements

1. Miva Merchant 5.5
2. MySQL backend - while the module may work on mivasql-based stores, it is not supported and may be subject to Application Timeout errors
3. Miva Merchant Engine v5.06 or higher (see note below about short URLs and token strings if you're using a version lower than 5.08)

Installation

Installing the Module

1. Unzip the module's ZIP file to your hard drive
2. Log into your Miva admin utility
3. Expand the navigation tree next to Global Settings on the left side of the screen
4. Click the link "[Add]" next to the "Modules" link
5. Click the "Upload" button on the right side of the screen
6. In the popup window, click the "Browse" button
7. Browse to the location where you stored the files from extracted from the ZIP file
8. Select NBFEEEDS5.mvc
9. Click the "Upload" button
10. When the module has been uploaded, the small "Upload File" window will disappear, leaving you with the main Miva admin screen visible
11. Click the "Add" button on the main Miva admin screen

Associating the Module with a Store

1. Under your store on the left side of the admin, click "Utilities"
2. In the module list, you should see an entry for "NetBlazon's Ultimate Feed Generator"
3. Check the checkbox next to this module and click the Update button

Entering your License Key

1. After assigning the module to the store, you should see a tab across the top of the right-hand screen in admin labeled "Ultimate Feed Generator"
2. Click the tab, and the module will present you with a form for entering your license key
3. Copy-and-paste your license key into the box and click the Update button

Usage

To use the Ultimate Feed Generator module, click the plus sign (+) next to Utilities on the left side of your Miva Merchant administration screen. Then click the Ultimate Feed Generator link in the section that opens up below Utilities. The main Ultimate Feed Generator screen will be displayed on the right side:

Ultimate Feed Generator [Documentation](#) | [FAQs](#) | [Support](#)

License Key: [REDACTED] [Check for Updates](#)
Version: 5.091 — [Check for Updates](#)
Copyright © 2002-2006 by [NetBlazon](#)

NETBLAZON

List All Feeds | **Module Settings**

Select?	ID	Feedname	Filename	Last Run	
<input type="checkbox"/>	1	FeedExact	feedexact.xml	11/16/2010 10:22	Run Feed Edit
<input type="checkbox"/>	2	XML Sitemap - Google, Yahoo, and Bing	sitemap.xml	Never	Run Feed Edit
<input type="checkbox"/>	3	GoogleBase	googlebase.bt	11/16/2010 10:22	Run Feed Edit
<input type="checkbox"/>	4	PriceGrabber	pricegrabber.bt	Never	Run Feed Edit
<input type="checkbox"/>	5	Yahoo! Product Submit	data.bt	Never	Run Feed Edit

[Delete](#) [Run Selected](#)

The main screen lists the feed templates already defined in the module. For each template, you can see its numeric ID, the name of the feed, the filename generated, and the last time the feed was run. You can run a single feed by clicking on its “Run Feed” link, or you can generate multiple feeds by selecting their “Select?” checkboxes and then clicking the “Run Selected” button in the lower right corner.

To edit a feed, click its “Edit” link, and to delete a feed, check its “Select?” checkbox and then click the “Delete” button.

You can add a new template by clicking the “Add” button in the top right corner.

When you add or edit a new feed template, you’ll see the details screen:

Ultimate Feed Generator

License Key: XXXXXXXXXX
 Version: 5.091 — [Check for Updates](#)
 Copyright © 2002-2009 by [NetBlazon](#)

[Documentation](#) | [FAQs](#) | [Support](#)

NETBLAZON

[List All Feeds](#)

[Module Settings](#)

Template for Feed GoogleBase [Run feed](#)

Feed Name *

File Name *

Last Run

Copy Completed Feed File to:

Template:

```

description*id*link*price*title*image_link*weight*condition
<xml:foreach iterator="product" array="products">
  <xml:product:description*><xml:product:code*><xml:global:domain
</xml:foreach>
    
```

Notes:

Versions: [Recall](#) [Clear History](#)

Exclude Inactive
Products/Categories?

☒

Exclude Out-of-Stock
Products?

☐

Include/Exclude Products

☒ Exclude these products ☐ Include only these products

Include/Exclude Categories

☒ Exclude these categories ☐ Include only these categories

Include/Exclude Products by
Category

☒ Exclude products in these categories ☐ Include only products in these categories

Maximum Price \$

Minimum Price \$

Maximum Cost \$

Minimum Cost \$

Maximum Weight

Minimum Weight

Category Short URL Format

Product Short URL Format

Token String 1 Format

Token String 2 Format

Token String 3 Format

Token String 4 Format

Load Attributes and Options?
(Slower)

☐

Load Custom Fields?
(Slower)

☒

Load Primary Category from
Category Breadcrumbs
Plus? (Slower)

☐

[Save](#)

There are a number of settings on this page.

Feed Name	The name you assign the feed – this is for descriptive purposes and is not used when generating the feed.
File Name	The name of the file that gets created when you generate the feed. This will be an actual file that is stored on your server and that can be downloaded to your harddrive, uploaded to comparison shopping engines, etc.
Last Run	This is not an editable field, but it lets you see when the feed was last generated. If it has not yet been generated, it will say <i>Never</i> .
Copy Completed Feed File To	By default, the feed files are placed in your exports directory (such as mivadata/Merchant5/s01/export). This location is typically not accessible by a web browser. If you want to copy the file to a web-accessible location after it's been generated, enter that value here. For example, enter / (a forward slash) to copy it to your root directory. Or enter /feedsoutput/ to copy the file to a folder called "feedsoutput". Then you could read the file by going to http://www.yourdomain.com/thefile.txt or http://www.yourdomain.com/feedsoutput/thefile.txt
Template	This is where you define what your template will generate when it's run against your store. Use Store Morph Technology for if statements and foreach loops, and to print out the values of variables such as the category name or product price. Use a "^" character to print a tab to the output file. For more details, see the Store Morph Technology section of this document.
Notes	Like many Miva Merchant templates, Ultimate Feed Generator stores a revision history of the template. Here you can give your revision a name. Leaving it blank will assign the current timestamp to the revision name.
Versions	Use this section to recall previous revisions of the template if needed.
Exclude Inactive Products/Categories?	Checking this box will exclude any products and categories that are marked as inactive in your store. Unchecking the box will include both active and inactive products and categories.
Exclude Out-of-Stock Products?	If checked, this excludes out-of-stock products using the built-in inventory system availability with Miva Merchant. It does not yet work with the new Wombat-enabled attribute-level inventory.
Include/Exclude Products	Enter a pipe-delimited list of product codes that you want to include – for example: apple banana pear. The radio button "Exclude these products" will build the file with all products except the ones listed here. The radio button "Include only these products" will build the file with only these products.
Include/Exclude Categories	Enter a pipe-delimited list of category codes that you want to include – for example: fruits vegetables. The radio button "Exclude these categories" will build the file with all categories except the ones listed here. The radio button "Include only these categories" will build the file

	with only these categories. PLEASE NOTE: this does not exclude or include PRODUCTS based on the categories they are assigned to – the next option (“Include/Exclude Products by Category”) does that. Users often get these two fields confused!
Include/Exclude Products by Category	Enter a pipe-delimited list of category codes that you want to include – for example: fruits vegetables. The radio button “Exclude products in these categories” will build the file with all products except the ones assigned to categories listed here. The radio button “Include only products in these categories” will build the file with only products that are assigned to these categories.
Maximum Price	If entered, products whose price is greater than this value will be excluded.
Minimum Price	If entered, products whose price is less than this value will be excluded.
Maximum Cost	If entered, products whose cost is greater than this value will be excluded.
Minimum Cost	If entered, products whose cost is less than this value will be excluded.
Maximum Weight	If entered, products whose weight is greater than this value will be excluded.
Minimum Weight	If entered, products whose weight is less than this value will be excluded.
The next group of fields are used to compensate for a bug in the SMT compiler that is part of Miva Empresa. You will need to use these fields if you want to create short links or other hardcoded URLs within your template, and you are using a version of Empresa lower than 5.08. To check what version you are using, click the “Main” link at the top of the left frame in your Miva Merchant administration, and then on the right side look for a line like “Miva Merchant Engine v5.06”. If the version is 5.08 or higher, you do not need to use these fields.	
Category Short URL Format	The short link format for your category page urls. An example would be <i>http://www.yourdomain.com/%catcode%.html</i> See the section “string tokens” for a list of available tokens. This can be accessed in the template with the entity <i>&mvt:category:shorturl;</i>
Product Short URL Format	The short link format for your product page urls. An example would be <i>http://www.yourdomain.com/%prodcode%.html</i> See the section “string tokens” for a list of available tokens. This can be accessed in the template with the entity <i>&mvt:product:shorturl;</i>
Token String 1 Format	An additional token string field that works like the short URL format fields for the purpose of creating additional hardcoded URLs in your feed files. For example, if you base your image filenames on your product code instead of specifying them in the database fields, you might set this to <i>http://www.yourdomain.com/images/%prodcode%.jpg</i> This can be accessed in the template with the entity <i>&mvt:product:tstr1;</i>
Token String 2 Format	A second token string; see Token String 1 Format. This can be accessed in the template with the entity <i>&mvt:product:tstr2;</i>
Token String 3 Format	A third token string; see Token String 1 Format. This can be accessed in the template with the entity <i>&mvt:product:tstr3;</i>

Token String 4 Format	A fourth token string; see Token String 1 Format. This can be accessed in the template with the entity <i>&mvt:product:tstr4;</i>
The remaining fields can be used no matter what version of Empresa your site uses.	
Load Attributes and Options?	Check this box if you want to load and use the attribute and option data for your products. Loading this data takes the feed longer to generate, so leave it unchecked if you do not need this data.
Load Custom Fields?	Check this box if you want to load and use the custom category and/or custom product field data. Loading this data takes the feed longer to generate, so leave it unchecked if you do not need this data.
Load Primary Category from Category Breadcrumbs Plus?	Check this box if you are using our Category Breadcrumbs Plus module and its primary category feature. It will load the primary category for each product. Loading this data takes the feed longer to generate, so leave it unchecked if you do not need this data.

Store Morph Technology

With your template definition, you should use the same Store Morph Technology used on the pages within Miva Merchant. You can use the pre-defined templates to get an idea of how this works within Ultimate Feed Generator. The main things you will use are *<mvt:if> ... <mvt:else>* tags and *<mvt:foreach>* loops, and entities such as *&mvt:product:name;*

Most likely in a feed template you will want to loop through your categories and/or products and print out information about each one. The two main loops you can use in a template look like this:

Category loop:

```
<mvt:foreach iterator="category" array="categories">
    &mvt:category:code;^&mvt:category:name;^etc...
</mvt:foreach>
```

Product loop:

```
<mvt:foreach iterator="product" array="products">
    &mvt:product:code;^&mvt:product:name;^etc...
</mvt:foreach>
```

The caret (^) character is a special character that will be replaced with a tab when the feed is generated. This is used to make tab-delimited text files. Inside each loop, you use entities such as *&mvt:product:code;* which are replaced with the actual code for each product. A full list of available entities is shown later in this document.

Within the product loop, you may want to loop through attributes and options. For example:

```
<mvt:foreach iterator="product" array="products">
    &mvt:product:code;^&mvt:product:name;^etc...
    <mvt:foreach iterator="attribute" array="product:attributes">
        <mvt:foreach iterator="option" array="attribute:options">
            &mvt:attribute:code;^&mvt:option:code;^etc.
        </mvt:foreach>
    </mvt:foreach>
</mvt:foreach>
```

Category Entities

&mvt:category:id;	Numeric category ID
&mvt:category:parent_id;	Numeric ID of this category's parent category
&mvt:category:parent_code;	Code of this category's parent category
&mvt:category:agrpcount;	Count of the number of availability groups the category is assigned to
&mvt:category:disp_order;	Numeric display order of the category within the store
&mvt:category:code;	Category code
&mvt:category:name;	Category name
&mvt:category:active;	1 or 0 depending on if category is active or not
&mvt:category:shorturl;	The short URL to the category page as defined in the short URL field (see explanation of token strings above where the admin settings screen is described)
&mvt:category:tstr1;	Category's token string #1 (see explanation of token strings above where the admin settings screen is described)
&mvt:category:tstr2;	Category's token string #2
&mvt:category:tstr3;	Category's token string #3
&mvt:category:tstr4;	Category's token string #2
&mvt:category:tree_image;	The category tree image (is not fully qualified – exactly as shown in database)
&mvt:category:title_image;	The category title image (is not fully qualified – exactly as shown in database)

Product Entities

&mvt:product:id;	Numeric product id
&mvt:product:catcount;	Count of the categories the product is assigned to
&mvt:product:agrpcount;	Count of the number of availability groups the product is assigned to
&mvt:product:pgrpcount;	Count of the number of price groups the

	product is assigned to
&mvt:product:disp_order;	Numeric display order of the product within the store
&mvt:product:code;	Product code
&mvt:product:name;	Product Name
&mvt:product:nameNoHTML;	Product Name with all HTML removed
&mvt:product:thumbnail;	Thumbnail as stored in the database (e.g. <i>graphics/00000001/thumbnail.jpg</i>)
&mvt:product:thumbnailFull;	Fully-qualified path to the thumbnail (e.g. <i>http://wwwyourdomain.com/mm5/graphics/00000001/thumbnail.jpg</i>)
&mvt:product:image;	Full-sized image as stored in the database (e.g. <i>graphics/00000001/image.jpg</i>)
&mvt:product:imageFull;	Fully-qualified path to the full-sized image (e.g. <i>http://wwwyourdomain.com/mm5/graphics/00000001/image.jpg</i>)
&mvt:product:price;	Product Price (no price groups or other adjustments made)
&mvt:product:cost;	Product Cost
&mvt:product:descrip;	Product Description as entered, including all line breaks and HTML
&mvt:product:descripNoHtml;	Product Description with all HTML and line breaks removed
&mvt:product:descripNoBreaks;	Product Description with line breaks removed, but without any HTML tags removed
&mvt:product:weight;	Product weight
&mvt:product:taxable;	1 or 0 depending on if product is taxable or not
&mvt:product:active;	1 or 0 depending on if product is active or not
&mvt:product:lastupdated;	When the product was last updated
&mvt:product:tstr1;	Product's token string #1 (see explanation of token strings above where the admin settings screen is described)
&mvt:product:tstr2;	Product's token string #2
&mvt:product:tstr3;	Product's token string #3
&mvt:product:tstr4;	Product's token string #4
&mvt:product:tstr1L;	Product's token string #1 converted to lower case

&mvt:product:tstr2L;	Product's token string #2 converted to lower case
&mvt:product:tstr3L;	Product's token string #3 converted to lower case
&mvt:product:tstr4L;	Product's token string #4 converted to lower case
&mvt:product:tstr1U;	Product's token string #1 converted to upper case
&mvt:product:tstr2U;	Product's token string #2 converted to upper case
&mvt:product:tstr3U;	Product's token string #3 converted to upper case
&mvt:product:tstr4U;	Product's token string #4 converted to upper case
&mvt:product:prodstats;	Number of times the product has been sold since the last time the store's statistics were reset
&mvt:product:rating;	Product rating (only available when Sebenza's Ultimate Ratings and Reviews is installed)
&mvt:product:reviewcount;	Number of product reviews (only available when Sebenza's Ultimate Ratings and Reviews is installed)
&mvt:product:inventory;	The actual number of items in inventory – may be positive or negative, or zero
&mvt:product:positiveInventory;	The actual number of items in inventory, where negative counts have been changed to zero (necessary for Amazon Seller Central feeds)
&mvt:product:shorturl;	The short URL to the product page as defined in the short URL field (see explanation of token strings above where the admin settings screen is described)
&mvt:product:catcodes;	A comma-delimited list of category codes
&mvt:product:catcodespipe;	A pipe-delimited list of category codes
&mvt:product:catids;	A comma-delimited list of numeric category IDs
&mvt:product:catidspipe;	A pipe-delimited list of numeric category IDs
&mvt:product:catnames;	A comma-delimited list of category names
&mvt:product:catnamespipe;	A pipe-delimited list of category names
&mvt:product:categorytrail;	A breadcrumb-style hierarchy based on the first category assignment in the database

	(such as Food > Fruits > Citrus)
&mvt:product:primarycategorycode;	The code of the primary category as specified in NetBlazon's Category Breadcrumbs Plus module
&mvt:product:primarycategoryname;	The name of the primary category as specified in NetBlazon's Category Breadcrumbs Plus module
&mvt:product:primarycategorybreadcrumbs;	A breadcrumb-style hierarchy based on the primary category as specified in NetBlazon's Category Breadcrumbs Plus module (such as Food > Fruits > Citrus)
&mvt:product:customfield_values:customfields:THECODE;	The value of the custom field with the code "THECODE"
&mvt:product:customfield_values:customfields_clean:THECODE;	The value of the custom field with the code "THECODE" with all HTML removed

Attribute Entities

&mvt:attribute:id	Numeric attribute id
&mvt:attribute:attemp_id	Numeric attribute template ID (only if it's a linked attribute template)
&mvt:attribute:default_id	ID of the default option
&mvt:attribute:disp_order	Display order of the attribute
&mvt:attribute:code	Attribute code
&mvt:attribute:type	Attribute type (text, select, checkbox, memo, radio)
&mvt:attribute:prompt	Attribute prompt
&mvt:attribute:price	Attribute price (only for text, checkbox, and memo types)
&mvt:attribute:cost	Attribute cost
&mvt:attribute:weight	Attribute weight
&mvt:attribute:required	1 or 0 depending on if attribute is required or not
&mvt:attribute:image	Attribute image

Option Entities

&mvt:option:id;	Numeric ID of the option
&mvt:option:disp_order;	Display order of the option
&mvt:option:code;	Option code
&mvt:option:prompt;	Option prompt
&mvt:option:price;	Option price
&mvt:option:cost;	Option cost
&mvt:option:weight;	Option weight
&mvt:option:image;	Option image

Support

For questions or problems with your Ultimate Feed Generator module please send email to support@netblazon.com or visit our helpdesk at <https://www.netblazon.com/support/helpdesk/>